awesome — since your tables are already added in Power Apps, here’s a **copy-paste, step-by-step** to turn them into a working KPI dashboard. I’ll assume your main table is named **KPITable** with columns:

Pillar, Objective, Priority, KPI, TimePeriod, KPI\_Targets, KPI\_Actuals

You can rename the data source in the formulas if yours is different.

**0) App bootstrap (thresholds + computed columns)**

**App → OnStart** (or Screen1 → OnVisible if you prefer)

// thresholds for status

Set(varGreenThreshold, 1.0); // 100% of target or more = Green

Set(varYellowThreshold, 0.8); // 80%–99% of target = Yellow

// bring data in and compute PerfPct + Status

ClearCollect(

colKPIs,

AddColumns(

KPITable,

PerfPct,

If(Value(KPI\_Targets)=0, Blank(), Value(KPI\_Actuals) / Value(KPI\_Targets)),

Status,

Switch(

true,

PerfPct >= varGreenThreshold, "Green",

PerfPct >= varYellowThreshold, "Yellow",

"Red"

)

)

);

// Pillar and TimePeriod pickers (with "All")

ClearCollect(colPillars,

Table({Value:"All"})

);

Collect(colPillars, ForAll(Sort(Distinct(colKPIs, Pillar), Result, Ascending), {Value: Result}));

ClearCollect(colPeriods,

Table({Value:"All"})

);

Collect(colPeriods, ForAll(Sort(Distinct(colKPIs, TimePeriod), Result, Ascending), {Value: Result}));

**Run** the OnStart (File → Run OnStart) once after pasting.

**1) Add the filter controls (top bar)**

* **Dropdown** ddPillar
  + *Items*:
  + colPillars
  + *DisplayFields / Value*: ["Value"]
  + *Default*:
  + LookUp(colPillars, Value="All")
* **Dropdown** ddPeriod
  + *Items*:
  + colPeriods
  + *DisplayFields / Value*: ["Value"]
  + *Default*:
  + LookUp(colPeriods, Value="All")
* **Combo box** cbStatus (multi-select)
  + *Items*:
  + ["Green","Yellow","Red"]
  + *SelectMultiple*: **true**
  + *DefaultSelectedItems* (optional):
  + ["Green","Yellow","Red"]
* **Text input** txtSearch
  + Placeholder: “Search KPI or Objective…”

**2) The filtered dataset (one formula to reuse everywhere)**

We’ll compute the filtered set **inline** wherever needed to keep it simple:

With(

{

ds: colKPIs

},

Filter(

ds,

// Pillar filter

(ddPillar.Selected.Value="All" || Pillar = ddPillar.Selected.Value)

&&

// Period filter

(ddPeriod.Selected.Value="All" || TimePeriod = ddPeriod.Selected.Value)

&&

// Status filter (multi-select)

(IsEmpty(cbStatus.SelectedItems) || Status in cbStatus.SelectedItems.Value)

&&

// Text search

(IsBlank(txtSearch.Text)

|| StartsWith(KPI, txtSearch.Text)

|| StartsWith(Objective, txtSearch.Text))

)

)

We’ll paste that into the **Items** property of the gallery and also reuse it for tiles.

**3) KPI gallery**

Insert a **Vertical Gallery** called galKPI.

* **Items**:
* SortByColumns(
* With(
* { ds:
* Filter(
* colKPIs,
* (ddPillar.Selected.Value="All" || Pillar = ddPillar.Selected.Value)
* && (ddPeriod.Selected.Value="All" || TimePeriod = ddPeriod.Selected.Value)
* && (IsEmpty(cbStatus.SelectedItems) || Status in cbStatus.SelectedItems.Value)
* && (IsBlank(txtSearch.Text) || StartsWith(KPI, txtSearch.Text) || StartsWith(Objective, txtSearch.Text))
* )
* },
* ds
* ),
* "Priority", // or "Pillar" then "Priority"
* Ascending
* )

Inside the gallery template add:

* **Label** lblKPI  
  *Text*:
* ThisItem.KPI
* **Label** lblMeta (small subtitle line)  
  *Text*:
* ThisItem.Pillar & " • " & ThisItem.Objective & " • " & ThisItem.TimePeriod
* **Label** lblValues (right-aligned)  
  *Text*:
* "Actual: " & Text(Value(ThisItem.KPI\_Actuals), "[$-en-US]#,##0.##") &
* " Target: " & Text(Value(ThisItem.KPI\_Targets), "[$-en-US]#,##0.##")

*Align*: **Right**

* **Rectangle** rectTrack (a light gray bar as the background of a progress bar)  
  *Fill*: RGBA(230,230,230,1)  
  *Width*: e.g., **300** (or stretch to layout)
* **Rectangle** rectProgress (placed on top of rectTrack, left-aligned)  
  *Width*:
* rectTrack.Width \* Min(1, Max(0, ThisItem.PerfPct))

*Fill*:

Switch(

ThisItem.Status,

"Green", Color.Green,

"Yellow", Color.Yellow,

"Red", Color.Red,

Color.Gray

)

* **Circle** icoLight (small traffic light)  
  *Fill*:
* Switch(
* ThisItem.Status,
* "Green", Color.Green,
* "Yellow", Color.Yellow,
* "Red", Color.Red,
* Color.Gray
* )

(Optional) Add a **sort toggle** icon icoSort with:

* *OnSelect*:
* Set(varSortAsc, !varSortAsc)
* Then change galKPI.Items to:
* SortByColumns(
* <same Filter With() as above>,
* "Priority",
* If(varSortAsc, Ascending, Descending)
* )

**4) Summary tiles (totals at top)**

Add **three Labels** (or small containers) and set their **Text**:

* **Total KPIs**
* CountRows(
* Filter(
* colKPIs,
* (ddPillar.Selected.Value="All" || Pillar = ddPillar.Selected.Value)
* && (ddPeriod.Selected.Value="All" || TimePeriod = ddPeriod.Selected.Value)
* && (IsEmpty(cbStatus.SelectedItems) || Status in cbStatus.SelectedItems.Value)
* && (IsBlank(txtSearch.Text) || StartsWith(KPI, txtSearch.Text) || StartsWith(Objective, txtSearch.Text))
* )
* )
* **On Track (Green)**
* CountIf(
* Filter(
* colKPIs,
* (ddPillar.Selected.Value="All" || Pillar = ddPillar.Selected.Value)
* && (ddPeriod.Selected.Value="All" || TimePeriod = ddPeriod.Selected.Value)
* && (IsEmpty(cbStatus.SelectedItems) || Status in cbStatus.SelectedItems.Value)
* && (IsBlank(txtSearch.Text) || StartsWith(KPI, txtSearch.Text) || StartsWith(Objective, txtSearch.Text))
* ),
* Status = "Green"
* )
* **Off Track (Red)**
* CountIf(
* Filter(
* colKPIs,
* (ddPillar.Selected.Value="All" || Pillar = ddPillar.Selected.Value)
* && (ddPeriod.Selected.Value="All" || TimePeriod = ddPeriod.Selected.Value)
* && (IsEmpty(cbStatus.SelectedItems) || Status in cbStatus.SelectedItems.Value)
* && (IsBlank(txtSearch.Text) || StartsWith(KPI, txtSearch.Text) || StartsWith(Objective, txtSearch.Text))
* ),
* Status = "Red"
* )

(Optional) **% On Track**

With(

{ f:

Filter(

colKPIs,

(ddPillar.Selected.Value="All" || Pillar = ddPillar.Selected.Value)

&& (ddPeriod.Selected.Value="All" || TimePeriod = ddPeriod.Selected.Value)

&& (IsEmpty(cbStatus.SelectedItems) || Status in cbStatus.SelectedItems.Value)

&& (IsBlank(txtSearch.Text) || StartsWith(KPI, txtSearch.Text) || StartsWith(Objective, txtSearch.Text))

)

},

Text( Round(CountIf(f, Status="Green") / Max(1, CountRows(f)) \* 100, 0), "0%" )

)

**5) Detail panel (right side) + small trend chart**

Add a **Container** as a side panel. Bind to galKPI.Selected.

* **Title label**:  
  *Text*:
* galKPI.Selected.KPI
* **Objective line**:
* galKPI.Selected.Objective
* **Target vs Actual**:
* "Target: " & Text(Value(galKPI.Selected.KPI\_Targets), "[$-en-US]#,##0.##")
* & " | Actual: " & Text(Value(galKPI.Selected.KPI\_Actuals), "[$-en-US]#,##0.##")
* **Status chip** (small label with fill color)
  + *Fill*:
  + Switch(
  + galKPI.Selected.Status,
  + "Green", Color.Green,
  + "Yellow", Color.Yellow,
  + "Red", Color.Red,
  + Color.Gray
  + )
  + *Text*: galKPI.Selected.Status
* **Line chart** (optional sparkline by TimePeriod)
  + Data source is the same table; if you have **multiple rows per KPI across periods**, set chart’s **Items** to:
  + SortByColumns(
  + Filter(colKPIs, KPI = galKPI.Selected.KPI && Pillar = galKPI.Selected.Pillar),
  + "TimePeriod",
  + Ascending
  + )
  + Map **X** to TimePeriod, **Y** to KPI\_Actuals.  
    If TimePeriod is text like “2025-09”, you can compute a sortable date:
  + AddColumns(
  + Filter(colKPIs, KPI = galKPI.Selected.KPI && Pillar = galKPI.Selected.Pillar),
  + XDate, DateValue(TimePeriod)
  + )

Then sort by XDate and bind X to XDate.

**6) Edit & save (optional)**

If you want to let owners **update** KPI\_Actuals or KPI\_Targets:

* Insert **Edit form** frmKPI
  + *DataSource*: KPITable
  + *Item*: galKPI.Selected
  + Include fields: KPI\_Targets, KPI\_Actuals
* Add a **Save** button:
  + *OnSelect*:
  + SubmitForm(frmKPI);
  + // refresh in-memory collection so Status/PerfPct recalc
  + ClearCollect(
  + colKPIs,
  + AddColumns(
  + KPITable,
  + PerfPct,
  + If(Value(KPI\_Targets)=0, Blank(), Value(KPI\_Actuals) / Value(KPI\_Targets)),
  + Status,
  + Switch(
  + true,
  + PerfPct >= varGreenThreshold, "Green",
  + PerfPct >= varYellowThreshold, "Yellow",
  + "Red"
  + )
  + )
  + );

⚠️ If your source is **Excel in OneDrive/SharePoint**, multi-user edits can be flaky. For production, prefer **SharePoint list, Dataverse, or SQL**.

**7) Nice touches (optional, quick)**

* **Priority color dot** in gallery:
* Switch(
* ThisItem.Priority,
* "High", Color.OrangeRed,
* "Medium", Color.Orange,
* "Low", Color.DarkSeaGreen,
* Color.Gray
* )
* **KPI card border color** by status:
* Switch(
* ThisItem.Status,
* "Green", ColorValue("#1E8449"),
* "Yellow", ColorValue("#B7950B"),
* "Red", ColorValue("#922B21"),
* RGBA(200,200,200,1)

)